

Задачи для 1 курса (2012-2013 уч. год.)

3. Задачи на сортировку массивов

При решении следующих задач нужно реализовать отдельную функцию для сортировки массива чисел, функцию для проверки массива на упорядоченность, функцию для чтения массива из файла и функцию для генерирования случайного массива указанной длины.

Нужно составить тест для проверки трудоемкости алгоритмов сортировки. Для этого реализуется один из методов “медленной” сортировки, один из методов “быстрой” сортировки и сортировка с использованием библиотечной функции `qsort`. Для заданного N генерируется неупорядоченный массив из N чисел, который сортируется всеми тремя методами. Замеряется время работы каждой сортировки (например, с помощью функции `clock()` или любых других подходящих). Тест повторяется несколько раз с удвоением длины массива N .

Алгоритмы медленной сортировки:

1. Простая сортировка обменов.
2. Пузырьковая сортировка.
3. Сортировка просеиванием.
4. Вставка с последовательным поиском.

Алгоритмы “быстрой” сортировки:

5. Сортировка слиянием.
6. Быстрая сортировка (quicksort).
7. Пирамидальная сортировка (heapsort).
8. Линейная сортировка (подсчетом) целого массива.
9. Сортировка целого массива группировкой с последовательным упорядочиванием битов.

В следующих задачах требуется реализовать сортировку с использованием “нетрадиционных” функций сравнения элементов массива.

10. Сгруппировать положительные элементы в начале массива, а отрицательные в конце так, чтобы и положительные, и отрицательные элементы внутри своей группы были упорядочены по возрастанию.
11. В данном массиве целых чисел упорядочить по возрастанию элементы с четными значениями, а элементы с нечетными значениями оставить на их исходных местах.
12. Упорядочить элементы массива целых чисел по возрастанию количества единиц в их битовом представлении.

4. Задачи на битовые операции и делимость

1. Написать функции, записывающие 0 или 1 в указанный бит данного целого числа и оставляющие остальные биты без изменения.
2. Проверить четность количества единиц в двоичном представлении данного целого числа.
3. Найти первые N целых чисел, у которых младший байт является зеркальным отражением следующего байта.
4. Определить позицию самой старшей единицы в битовом представлении данного целого числа.
5. Написать функции, позволяющие работать с целым числом (`long`) как с “массивом” четырех байтовых чисел (`char`).
6. Написать функцию, которая зеркально переворачивает битовое представление целого числа (аналогично перестановке массива в обратном порядке).
7. Написать функцию, которая циклически сдвигает битовое представление целого числа на указанное количество позиций вправо или влево (аналогично циклическому сдвигу массива).
8. Написать функцию, которая переставляет байты целого числа (`int`) в обратном порядке.
9. Возвести число в степень N за не более чем $2 \log_2 N$ умножений.
10. Вывести в файл все подмножества множества $\{1, \dots, N\}$.
11. Вывести в файл все k -элементные подмножества мн-ва $\{1, \dots, N\}$.
12. Найти наибольший общий делитель двух целых чисел (алгоритм Евклида).
13. Определить четность произвольной перестановки N чисел.
14. Вычислить первые N простых чисел.
15. Разложить натуральное число на простые множители.
16. Вывести значение целого числа N в “словесной форме”.
17. Вычислить представление числа $1/N$ в виде десятичной дроби (начало и период) (то же для числа M/N).

5. Задачи на обработку множества точек

В следующих задачах предполагается, что в файле записано несколько пар чисел, которые можно рассматривать как координаты множества точек на плоскости или как координаты множества концов отрезков на прямой.

1. Множество точек определяет ломаную. Имеет ли она самопересечения?
2. Множество точек определяет многоугольник. Является ли он выпуклым?
3. Множество точек определяет многоугольник. Определить угол, под которым данный многоугольник виден из заданной точки.
4. Множество точек определяет многоугольник. Для данной точки определить где она расположена относительно этого многоугольника: внутри, снаружи, на границе.
5. Множество точек определяет выпуклый многоугольник. Найти его минимальный и максимальный диаметры (вписанная и описанная окружности).
6. Дано множество отрезков на прямой. Покрывает ли объединение этих отрезков заданный отрезок $[a, b]$?
7. Два множества точек задают два многоугольника (без самопересечений, но не обязательно выпуклые). Определить расстояние между этими многоугольниками.
8. Два множества точек задают два выпуклых многоугольника, не лежащих один внутри другого. Определить расстояние между этими многоугольниками с линейной оценкой трудоемкости по суммарному количеству вершин.
9. Дано множество точек. Найти центр и радиус минимального круга, который содержит все эти точки.
10. Дано множество отрезков на прямой. Выбрать из него и вывести те отрезки, объединение которых дает отрезок наибольшей длины.
11. Даны центры равномерно растущих кругов на плоскости. При столкновении друг с другом столкнувшиеся круги прекращают свой рост. Найти радиусы кругов, когда процесс роста остановится полностью.
12. Дано множество точек на плоскости. Построить выпуклую оболочку этого множества.
13. Множество точек определяет выпуклый многоугольник. Построить многоугольник, который получится, если линию, задающую каждую сторону, отодвинуть в перпендикулярном ей направлении на величину h .